Analysis of IP-based Real-Time In-Car Networks with Network Calculus

Revsbech, Kasper; Nielsen, Jimmy Jessen; Harritsø, Kevin; Steffen, Rainer

Publication date:
2009

Document Version
Publisher's PDF, also known as Version of record

Analysis of IP-based Real-Time In-Car Networks with Network Calculus

Kasper Revsbech, Kevin Harritsø, Jimmy Jessen Nielsen
Electronic Systems
Aalborg University, Denmark
Email: {revsbech,khar04,jjn}@es.aau.dk

Rainer Steffen
BMW Group Research and Technology
Munich, Germany
Email: rainer.steffen@bmw.de

***Abstract*—This work investigates the possibility of using Deterministic Network Calculus to verify real-time constraints in IP based in-car networks. A combination of the Real-Time Calculus toolbox [?] and the Cyclic Network Calculus toolbox [?] has been used to create a model of a switch providing fixed priority/FIFO scheduling. Considerations on how to model such a switch in general and how to create a model in the toolbox are presented. The switch model is validated by inspection and soundness check of results for varying model parameters.**

I. Introduction

Automobiles are nowadays designed with an increasing amount of Electronic Control Units (ECUs) to provide higher functionality in regards to safety, performance, entertainment and comfort. These ECUs receive sensory input, process it and transmit control output to other ECUs, actuators, transducers etc. placed at various positions in the automobile. Therefore, an in-car network is deployed to interconnect these components and functions. Typically, an in-car network is based on a number of different fieldbuses/networks, such as CAN, MOST, LIN and FlexRay.

Recently an interest of unification of network technology has emerged. The BMW Group has been investigating the feasibility of creating an IP based in-car network using prioritized Ethernet, 802.1P for media access control and the IP protocol for routing and delivery of data packets between nodes in the network [?]. As concluded in [?], it is possible to create such a network and a proof of concept implementation has been conducted to support this conclusion. To investigate the feasibility and verify that the technology of IP and Ethernet can be used in such time critical applications, an analysis of the real-time performance has to be conducted. One way of doing so is by use of Deterministic Network Calculus (DNC) constituting a theoretical (non) deterministic basis for determining worst case bounds on bandwidth usage, delay and buffering requirements throughout a network and thereby applicable for verifying the real-time performance of networked systems. DNC was first introduced in [**?**] and later extensively described in [**?**].

This work investigates the possibilities of using the theory of DNC and DNC based tools in the design and verification process of an in-car network. Here we presents the first step of this investigation process; namely the construction of a network model of an in-car network. First a study of an in-car network is described in Sec. II where the topology and components on the network model are identified. In Sec. III the theory of DNC is briefly presented, along with an introduction to the DNC based MATLAB® toolbox Cyclic Network Calculus (CyNC), which uses the Real-Time Calculus (RTC) Toolbox as a framework for doing DNC calculations. Since Ethernet/IP routers are the key scheduling elements in a unified in-car networks, their models become essential for the design process. The modelling of a fixed priority/FIFO switch is presented in Sec. IV along with various possibilities for representing the models in DNC/CyNC. In Sec. V a validation of the switch model is conducted by combining the switch model with a model of data in an in-car network and inspecting healthiness of results. This is done by inspecting the impact of prioritization and by varying the delay for a particular element and inspect the impact on the maximum delays. A comparison with real recorded data is not reported in this work, but planned as an activity for the near future. Finally a conclusion of the presented work is given and the future perspectives are presented.

II. In-car Network description

To create a DNC model of an in-car network, one has to be able to describe the network topology and the components within the network. Here we present a possible network structure of an in-car network. There are at least six different traffic domains on the in-car network, which can be classified as follows: Chassis, Comfort, Driver assistance, Entertainment, Infotainment and Power train. Each of these domains consists of multiple nodes in which numerous connections of different priority are created. The traffic is then sent through a number of switches before reaching the destination.

However, instead of creating one large model of the network, the internals of each domain can be modelled separately as long as the traffic pattern leaving a domain remains the same, internal changes do not affect the other domains. This means that a topology as in Fig. 1 can be used where a central unit switch connects all the sub domains that can be modelled individually. However to do so, a model of the central unit switch has to be created. As described in [?] a solution to ensure the real-time demands of critical traffic is to use fixed prioritization, which means that the central unit switch is considered as a prioritized switch. It has been chosen to assign the priorities as listed in Table I to the flows traversing the

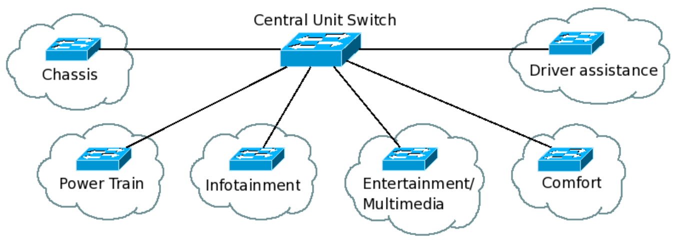

Fig. 1. The topology of an in-car network where a central unit switch is used to allow intercommunication between the six traffic domains

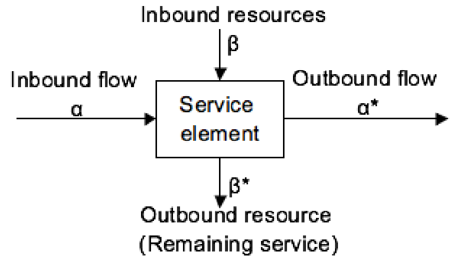

Fig. 2. The inputs and outputs for a service element. As shown the service element provides bounds on the outbound flow α^* and the outbound resource β^* which is the service left after serving flow α.

central unit switch. These priorities are chosen as realistic priorities but only serve the purpose of allowing test of the model and have to be reconsidered.

TABLE I
PRIORITIES OF THE TRAFFIC FLOWS TRAVERSING THE CENTRAL UNIT SWITCH. INTRA DOMAIN TRAFFIC IS NOT CONSIDERED IN THE DEFINITION OF THE PRIORITIES, AND IS THEREBY LEFT BLANK IN THE TABLE.

Priority	**Source**					
Destination	Chas.	Comf.	D. A.	Info.	Mult.	P. T.
Chassis		3	2	6	7	1
Comfort	3		5	6	7	4
Driver A.	2	5		6	7	2
Infotainment	6	6	6		6	5
Multimedia	7	7	7	7		5
Power T.	1	4	5	6	7	

III. Deterministic Network Calculus Method and Tools

In network calculus all traffic and service in the described network is expressed non-deterministically in terms of time dependent bounds. The traffic is expressed by upper and lower arrival bounds and the processing/communication-services by upper and lower service bounds. Arrival and service bounds are given in terms of time functions/curves bounding arrival/service above and below for a time interval of specified duration. In this work we confine ourselves to *affine* and *staircase* functions to represent arrival/service bounds. Eq. (1) presents an *affine arrival curve* where r denotes maximum long term arrival rate of the flow and b denotes the maximum transient burst of the flow. The function to express a *periodic arrival curve* can be seen in Eq. (2) where T denotes the interval between arrivals for the flow and τ the maximum delay jitter of periodic arrivals [?].

$$\gamma_{r,b}(t) = rt + b \qquad \forall \qquad t \geq 0 \tag{1}$$

$$\upsilon_{T,\tau}(t) = \lceil \frac{t+\tau}{T} \rceil \qquad \forall \qquad t \geq 0 \tag{2}$$

$$\beta_{r,T} = r[t-T]^+ \qquad \forall \qquad t \geq 0 \tag{3}$$

To express the service of a serving unit a so called *rate latency function* in Eq. (3) is used where r denotes the minimum long term service rate provided by the serving element and T the maximum latency time. As shown in Fig. 2 bounds based on the inbound flow and resources can be calculated per service element. Using the theory of [?] the outbound flow α^* can be found as $\alpha^* = \alpha \oslash \beta$ where α denotes the arrival curve that constrains the inbound flow and β denotes the service curve that constrains the inbound resources. To calculate the worst case queue length Q at the service element the maximal vertical distance between the arrival curve and the service curve is found as expressed in Eq. (4). The maximum delay D imposed by the service element is the maximal horizontal distance between the arrival and service curve and is found as expressed in Eq. (5) [?].

$$Q \leq \sup_{t\geq 0}\{\alpha(t) - \beta(t)\} \tag{4}$$

$$D \leq \sup_{t\geq 0}\{\inf\{\tau \mid \alpha(t) \leq \beta(t+\tau)\}\} \tag{5}$$

The theory of DNC is applied to the in-car network model with a combination of the CyNC toolbox [?] and the RTC toolbox [?] merged into one common toolbox using the calculation engine for e.g sup-minus convolution and expression of flows as curves from RTC combined with the graphical user interface, the ability to handle cyclic flows and the scheduling element algorithms from CyNC. This combination enables a graphical modeling tool where graphical blocks are used to express the arrival bounds and scheduling elements as well as connections between blocks to represent the routing in the network. Furthermore the properties of each element have to be specified. E.g. for a service/processing element the appropriate scheduling algorithm is specified along with service rate and preemption delay.

When a model has been created and the calculation engine has converged, the tool provides stability guarantees for the system, i.e. that every queue has an upper limit below ∞. Furthermore the tool calculates maximum queue length of a service element and the maximum delay that a traffic flow can experience for each service element. This enables a the user to calculate the total delay a flow can experience throughout the system. First the total service provided to the flow after traversing each service element is calculated by a convolution of each service provided, then Eq. (5) is used where β is the total service and α the arrival bound of the flow. By providing a stability guarantee, a guarantee of max delay and queue length the tool can be used to analyse the feasibility of a

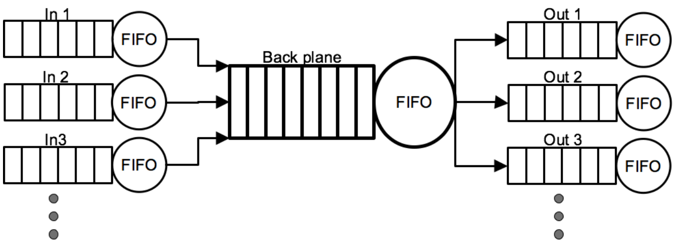

Fig. 3. A switch consisting of three types of service elements. Input ports, a back plane and output ports.

particular network consisting of a number of service elements and traffic flows.

IV. Switch modeling

The first step of investigating DNC for an in-car network is to create a general model of the central unit switch shown in Fig. 1. In this section we will show the principles of how a model of a prioritized switch can be constructed but without considering specific switch brand dependent details. In general at least three types of service elements exist within a switch: input ports, a back plane and output ports see Fig. 3. Note that the switch is modelled as full duplex meaning that every port has an input and an output queue that is independent but shares the same back plane. Each element can then be expressed by the appropriate scheduling method e.g. First In First Out (FIFO). As described in Sec. II a prioritizing setup with fixed priorities is considered. The CyNC toolbox provides a fixed priority block as default where a total service of the element is given as input and every flow is connected in its respective priority. However in the considered use-case, equally prioritized flows with potentially different source and destination exist. Therefore these flows cannot be aggregated and fed to the same input. We propose to solve this by utilizing the fact that a service element outputs the remaining service β^* after serving the arrival flows, show in Fig. 2. By using a FIFO scheduling element for flows of same priority and letting the top priority FIFO ”use” the needed resources and passing the remaining resources to the second priority FIFO a fixed priority scheduler service element capable of handling multiple flows of same priorities can be modelled in CyNC see an example in Fig. 4.

Designing a fixed priority switch one has to ensure that the priority is kept throughout the whole switch, implementing only the back plane as prioritized is not sufficient. The output ports must also have a queue for each priority. Of course the model also have to afflict the design which leads into a more complex model of the switch but also more precise, where the high priority flows do not have to suffer from the lower prioritized flows. Naturally one has to ensure that the switch in question actually is implemented in that way. A fully prioritized switch model can be seen in Fig. 5. By using this paradigm a switch providing fixed priority scheduling among the flows and FIFO for flows of same priority can be modelled.

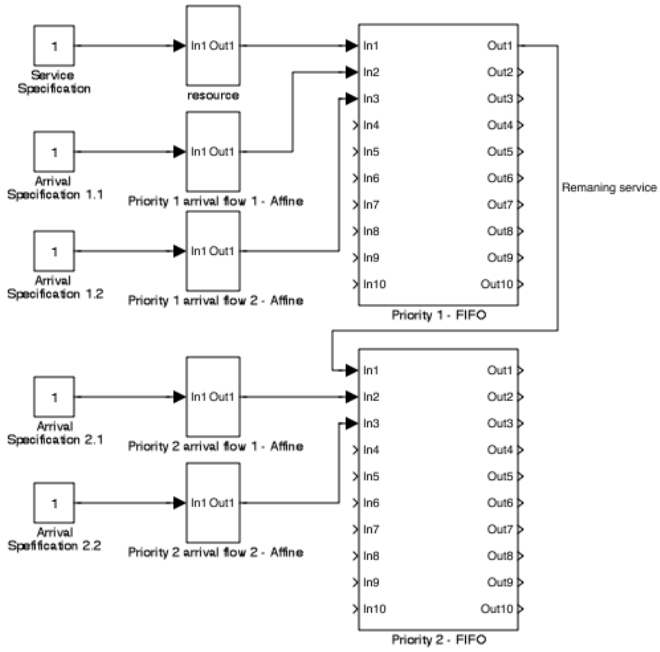

Fig. 4. A model of a Fixed priority scheduler with multiple flows of same priority in CyNC

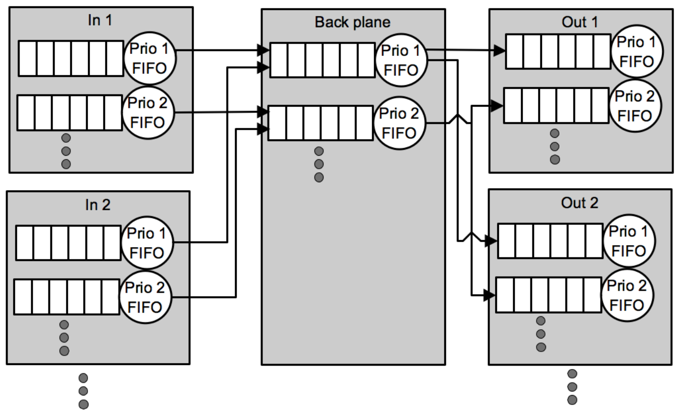

Fig. 5. A switch providing fixed priority scheduling where every in and out port has a queue for every priority, the back plane also have a queue for every priority

Using the concept shown in Fig. 4 one can build the model in CyNC.

V. Switch model validation

To validate the developed model of a fixed priority switch described in Section IV and to investigate the consequences of applying realistic data to the fixed priority central switch shown in Fig. 1, a CyNC model has been constructed based on the paradigm shown in Fig. 5. As shown in Fig. 1 there are six traffic domains each generating and receiving traffic. Therefore the CyNC model comprises six input and output ports. The properties of each scheduling element do not conform to any particular off-the-shelf switch, but are chosen within realistic values. It is chosen to set the minimum service

TABLE II

MAXIMUM DELAYS FOR THE FLOWS OF DIFFERENT PRIORITIES LEAVING THE CHASSIS DOMAIN. CALCULATED FOR TWO SETTINGS OF THE BACK PLANE ONE WHERE THE MAXIMUM DELAY OF SERVICE IS 0.053 MS AND ONE WITH 1.053 MS MAXIMUM DELAY

Flow	**Priority**	**Delay 0.053 ms**	**Delay 1.053 ms**
chassis→pTrain	1	0.1695 ms	1.169 ms
chassis→driverAssist	2	0.3024 ms	1.303 ms
chassis→comfort	3	0.3771 ms	1.378 ms

of the scheduling elements as follows: Input and output ports: 100 Mbit/Sec with no delay, back plane: 304 Mbit/Sec and max delay of 0.053 millisecond.

To validate the prioritization model prioritized traffic has to be applied. In this case it is chosen to apply the traffic specifications of an existing BMW configuration to ensure realistic arrival data[1]. This data arrives from the six domains shown in Fig. 1. The original flow data are derived from a repository of individual flows each represented by individual staircase arrival curves (Expressed by Eq. (2)) which give a very complex model in terms of creating an entity for every flow. To reduce the complexity flows sharing source, destination, receiver and priority have been aggregated. However this yields a complex arrival curve description and in turn computational complexity. To reduce the arrival curve complexity the aggregated arrival curve has been transformed into an affine arrival curve. The price of doing so is a slight overestimation of the arriving traffic which leads into an overestimate of the needed resources.

Using the presented switch model with the described data flows the maximum delays listed in Table II were obtained. As seen in the table the maximum delays for the flows leaving the Chassis are increasing as the priority gets lower. This is an immediate consequence of the fact that the lower priority flows have less resources. If the delay is increased in the back plane from 0.053 ms to 1.053 ms while keeping other settings fixed, the maximum delay becomes as listed in Table II. As seen in the table the total delays are only affected by the extra delay added to the back plane and as shown all the flows are affected by the reduced overall service service in the back plane.

VI. CONCLUSION AND OUTLOOK

The theory of DNC has been used to develop a fixed priority switch paradigm, which has been implemented in the matlab toolbox CyNC. Utilizing the knowledge of an in-car network, the developed switch paradigm and CyNC implementation have been used to model a fixed priority switch. The fixed priority switch model uses a chain of FIFO blocks to allow multiple equally prioritized input flows that cannot be aggregated. Test results have shown that the maximum delays increase for a flow as the priority is lowered.

[1]Note that even thought the input data is authentic the results do not show anything about the present and future cars networks as the properties of the scheduling elements only are realistic but not derived from equipment used in cars

The proposed modeling framework based on CyNC and DNC allows for direct specification of switch properties and inbound data flows, while the maximum delays can be read directly from the model. With the implementation of the fixed priority switch model, the first step towards an analytical verification and design method for IP based real-time in-car networks as proposed in [?] has been taken.

A next step could be to provide automated model data acquisition, i.e. for network components and traffic characteristics as well as preparation of model data for use in a DNC framework. Regarding the switch, the first job is to quantify its precise properties in the time domain. Model data may be extracted from data sheets, which however is often quite laborious and not feasible for automation. As an alternative, empirical methods based on system identification may be applied. Parameters for models in a repository of assumed internal structure model parameters may be found from observations of ingress and egress flows. Furthermore, the calculated delays and queue lengths of an existing system should be compared to the measured delays and queue lengths in an existing in-car network to estimate the overestimation imposed by a DNC based analysis.

ACKNOWLEDGMENT

The authors would like to thank BMW Group for the cooperation and supervision. We would also like to thank Johan de Claville Christiansen, Kristian Engh Lundgreen and Youssef El Otmani for the research effort behind this paper. The research work was in parts conducted in the context of a student project at the Department of Electronic Systems at Aalborg University, and as such supported by the Study Board at Aalborg University, see www.esn.aau.dk.